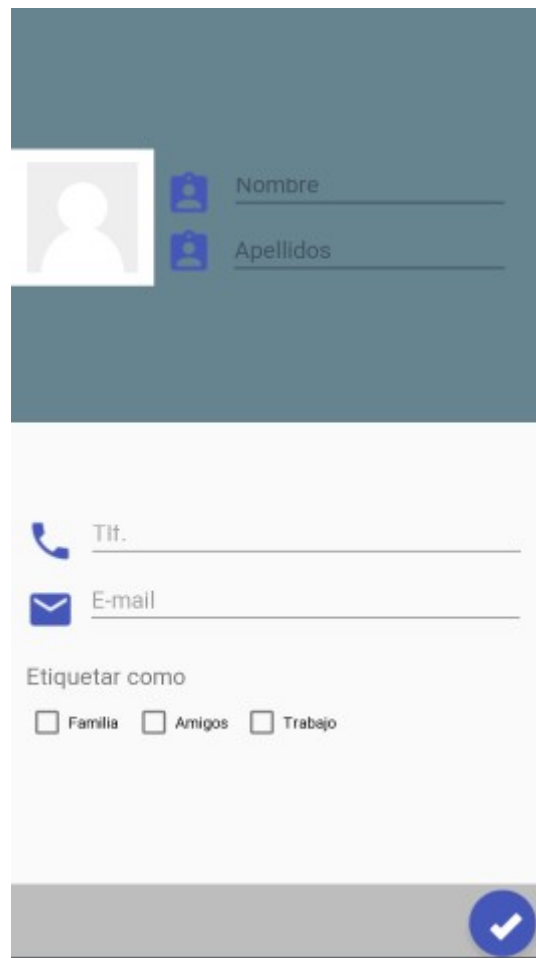


Vamos a realizar una actividad de Agenda que tendrá el siguiente aspecto:



The form is divided into two main sections. The top section has a dark blue background and contains a profile picture placeholder (a white silhouette on a grey square) and two input fields for 'Nombre' and 'Apellidos', each preceded by a small blue icon of a person. The bottom section has a light grey background and contains two input fields for 'Tlf.' and 'E-mail', each preceded by a blue icon of a telephone and an envelope respectively. Below these is a section titled 'Etiquetar como' with three checkboxes labeled 'Familia', 'Amigos', and 'Trabajo'. At the bottom right of the form is a blue circular button with a white checkmark.

Nombre _____

Apellidos _____

Tlf. _____

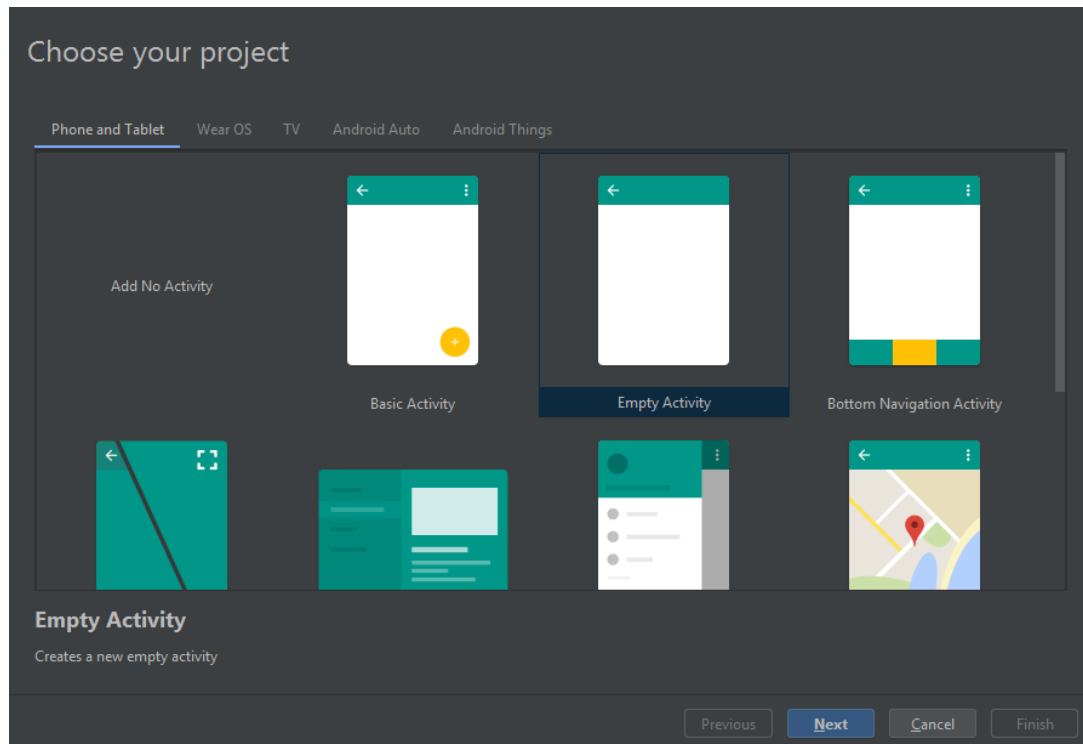
E-mail _____

Etiquetar como

☐ Familia ☐ Amigos ☐ Trabajo

☒

Primero de todo elegiremos el “Empty Activity” para empezar nuestro proyecto.



Elegiremos un constraint Layout, pero se puede elegir cualquier Layout para hacer esta actividad y en este layout primero de todo le pongo 2 frameLayout para colocar lo encima de la pantalla (uno más grande y el otro más pequeño en el que va a esta nuestra imagen de la agenda)

```
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".MainActivity">

    <FrameLayout
        android:id="@+id/frameLayout"
        android:layout_width="419dp"
        android:layout_height="320dp"
        android:background="@android:color/system_notification_accent_color"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent" />

    <FrameLayout
        android:id="@+id/frameLayout2"
        android:layout_width="109dp"
        android:layout_height="104dp"
        android:layout_marginStart="8dp"
        android:layout_marginLeft="8dp"
        android:layout_marginTop="8dp"
        android:background="@color/cardview_light_background"
        app:layout_constraintBottom_toBottomOf="@+id/frameLayout"
        app:layout_constraintEnd_toEndOf="@+id/frameLayout"
        app:layout_constraintHorizontal_bias="0.0"
        app:layout_constraintStart_toStartOf="@+id/frameLayout"
        app:layout_constraintTop_toTopOf="@+id/frameLayout" />

</FrameLayout>
```

Dentro del pequeño `FrameLayout` le ponemos la imagen, y la imagen de los nombres y la celda en el que vamos a escribir, le ponemos después.



Fijamos la parte izquierda de lo que tiene el aspecto.

La foto que vemos dentro del `FrameLayout` (pequeño) blanco es un “`imageView`” de abajo. Y le añadimos un `Floating action button`, no preocupamos del código donde está puesto cada elemento, porque le asignamos a cada elemento donde tiene que colocar para que aparezca un aspecto similar que nuestra agenda.

```
<ImageView
    android:id="@+id/imageView6"
    android:layout_width="85dp"
    android:layout_height="84dp"
    app:layout_constraintBottom_toBottomOf="@+id/frameLayout2"
    app:layout_constraintEnd_toEndOf="@+id/frameLayout2"
    app:layout_constraintStart_toStartOf="@+id/frameLayout2"
    app:layout_constraintTop_toTopOf="@+id/frameLayout2"
    app:srcCompat="@drawable/perfil" />

<com.google.android.material.floatingactionbutton.FloatingActionButton
    android:id="@+id/floatingActionButton"
    android:layout_width="58dp"
    android:layout_height="51dp"
    android:layout_marginEnd="8dp"
    android:layout_marginRight="8dp"
    android:clickable="true"
    android:src="@drawable/check_icon"
    app:backgroundTint="@color/design_default_color_primary"
    app:backgroundTintMode="src_in"
    app:fabSize="auto"
    app:layout_constraintBottom_toBottomOf="@+id/toolbar"
    app:layout_constraintEnd_toEndOf="@+id/toolbar"
    app:layout_constraintTop_toTopOf="@+id/toolbar" />
```

La `toolbar` lo ponemos luego. Y ponemos el mismo `id` que ya tiene puesto el `floatingActionButton`



Ponemos el texto “Etiquetar como” con sus propiedades correspondientes, y empezamos a poner nuestros checkBox.

```
<TextView
    android:id="@+id/textView"
    android:layout_width="129dp"
    android:layout_height="33dp"
    android:layout_marginStart="16dp"
    android:layout_marginLeft="16dp"
    android:layout_marginTop="16dp"
    android:text="Etiquetar como"
    android:textSize="18dp"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toBottomOf="@+id/imageView9" />

<CheckBox
    android:id="@+id/checkBox"
    android:layout_width="81dp"
    android:layout_height="29dp"
    android:layout_marginStart="16dp"
    android:layout_marginLeft="16dp"
    android:text="Familia"
    android:textSize="12dp"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toBottomOf="@+id/textView" />
```

y intentamos a conseguir hacerlo así como lo tenemos abajo:



Etiquetar como

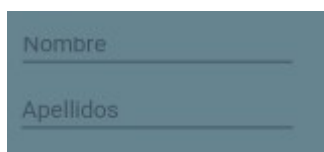
☐ Familia ☐ Amigos ☐ Trabajo

Así que ponemos 2 checkBox más y añadimos sus constraint Layout y propiedades correspondiente.

```
<CheckBox
    android:id="@+id/checkBox2"
    android:layout_width="74dp"
    android:layout_height="29dp"
    android:text="Amigos"
    android:textSize="12dp"
    app:layout_constraintStart_toEndOf="@+id/checkBox"
    app:layout_constraintTop_toBottomOf="@+id/textView" />

<CheckBox
    android:id="@+id/checkBox3"
    android:layout_width="78dp"
    android:layout_height="29dp"
    android:layout_marginStart="8dp"
    android:layout_marginLeft="8dp"
    android:layout_marginEnd="8dp"
    android:layout_marginRight="8dp"
    android:text="Trabajo"
    android:textSize="12dp"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.0"
    app:layout_constraintStart_toEndOf="@+id/checkBox2"
    app:layout_constraintTop_toBottomOf="@+id/textView" />
```

Y ahora empezamos a poner nuestro “EditText” en el que podamos escribir para cada cosa que nos pide en la agenda.

Un formulario de entrada de datos con un fondo gris azulado. Contiene dos campos de texto con bordes blancos y líneas inferiores. El primer campo está etiquetado con el texto "Nombre" en un color gris oscuro. El segundo campo está etiquetado con el texto "Apellidos" en el mismo color gris oscuro.

Vamos Avanzando con nuestro código, No pasa nada si nos sale algunas partes en roja, Mientras poniendo nuestra ruta de los identificadores, porque está un poco movido el código.

Para nuestro nombre y apellido:

```
<EditText
    android:id="@+id/editText"
    android:layout_width="212dp"
    android:layout_height="37dp"
    android:layout_marginStart="8dp"
    android:layout_marginLeft="8dp"
    android:layout_marginTop="124dp"
    android:layout_marginEnd="8dp"
    android:layout_marginRight="8dp"
    android:ems="10"
    android:hint="Nombre"
    android:inputType="textPersonName"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.2"
    app:layout_constraintStart_toEndOf="@+id/imageView7"
    app:layout_constraintTop_toTopOf="@+id/frameLayout" />

<EditText
    android:id="@+id/editText2"
    android:layout_width="213dp"
    android:layout_height="42dp"
    android:layout_marginTop="8dp"
    android:layout_marginEnd="28dp"
    android:layout_marginRight="28dp"
    android:ems="10"
    android:hint="Apellidos"
    android:inputType="textPersonName"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintTop_toBottomOf="@+id/editText" />
```

Avanzamos, y Empezamos a poner nuestra imagenes. Y las propiedades y layout_constraintStart_to startof etc. Correspondiente.

```
<ImageView
    android:id="@+id/imageView7"
    android:layout_width="37dp"
    android:layout_height="44dp"
    android:layout_marginStart="8dp"
    android:layout_marginLeft="8dp"
    android:layout_marginTop="124dp"
    android:tint="@color/design_default_color_primary"
    app:layout_constraintStart_toEndOf="@+id/frameLayout2"
    app:layout_constraintTop_toTopOf="@+id/frameLayout"
    app:srcCompat="@drawable/ic_cliente" />

<ImageView
    android:id="@+id/imageView9"
    android:layout_width="37dp"
    android:layout_height="44dp"
    android:layout_marginStart="16dp"
    android:layout_marginLeft="16dp"
    android:layout_marginTop="8dp"
    android:autofillHints="E-mail"
    android:tint="@color/design_default_color_primary"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toBottomOf="@+id/imageView12"
    app:srcCompat="@drawable/ic_correo" />
```

```
<ImageView
    android:id="@+id/imageView12"
    android:layout_width="37dp"
    android:layout_height="44dp"
    android:layout_marginStart="16dp"
    android:layout_marginLeft="16dp"
    android:layout_marginTop="68dp"
    android:tint="@color/design_default_color_primary"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toBottomOf="@+id/frameLayout"
    app:srcCompat="@drawable/ic_telefono" />

<ImageView
    android:id="@+id/imageView11"
    android:layout_width="37dp"
    android:layout_height="44dp"
    android:layout_marginStart="8dp"
    android:layout_marginLeft="8dp"
    android:layout_marginEnd="8dp"
    android:layout_marginRight="8dp"
    android:tint="@color/design_default_color_primary"
    app:layout_constraintEnd_toStartOf="@+id/editText2"
    app:layout_constraintHorizontal_bias="0.0"
    app:layout_constraintStart_toEndOf="@+id/frameLayout2"
    app:layout_constraintTop_toBottomOf="@+id/imageView7"
    app:srcCompat="@drawable/ic_cliente" />
```

Ponemos el “EditText”

```
<EditText
    android:id="@+id/editText3"
    android:layout_width="333dp"
    android:layout_height="38dp"
    android:layout_marginStart="8dp"
    android:layout_marginLeft="8dp"
    android:layout_marginTop="68dp"
    android:layout_marginEnd="16dp"
    android:layout_marginRight="16dp"
    android:ems="10"
    android:hint="Tlf."
    android:inputType="phone"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="1.0"
    app:layout_constraintStart_toEndOf="@+id/imageView12"
    app:layout_constraintTop_toBottomOf="@+id/frameLayout" />

<EditText
    android:id="@+id/editText4"
    android:layout_width="333dp"
    android:layout_height="43dp"
    android:layout_marginStart="8dp"
    android:layout_marginLeft="8dp"
    android:layout_marginTop="8dp"
    android:layout_marginEnd="16dp"
    android:layout_marginRight="16dp"
    android:ems="10"
    android:hint="E-mail"
    android:inputType="textEmailAddress"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintStart_toEndOf="@+id/imageView9"
    app:layout_constraintTop_toBottomOf="@+id/editText3" />
```

esto es lo que se añade visualmente.



Visual representation of the EditText fields. The first field is labeled "Tlf." with a phone icon. The second field is labeled "E-mail" with an email icon.

Al final le ponemos un toolbar sobre toolbar le colocamos nuestro floatingAction button, y el fondo le ponemos medio transparente.

```
<androidx.appcompat.widget.Toolbar
    android:id="@+id/toolbar"
    android:layout_width="411dp"
    android:layout_height="60dp"
    android:background="#42000000"
    android:minHeight="?attr/actionBarSize"
    android:theme="?attr/actionBarTheme"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="1.0"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toBottomOf="@+id/checkbox2"
    app:layout_constraintVertical_bias="1.0" />

</androidx.constraintlayout.widget.ConstraintLayout>
```

//El la parte de java he hecho un metodo en el que entra un booleano y una imagen, Y comprueba si el foco esta sobre el EditText, y se cambia el color de la imagen correspondiente, si se encuentra un foco en el EditText del mismo.

```
public void onFocusChangeParaTodos(boolean hasFocus, ImageView img) {
    ImageView ic = img;
    Drawable d = ic.getDrawable();
    d = DrawableCompat.wrap(d);
    if (hasFocus)
        DrawableCompat.setTint(d, ContextCompat.getColor(getApplicationContext(), R.color.design_default_color_primary));
    else
        DrawableCompat.setTint(d, ContextCompat.getColor(getApplicationContext(), R.color.design_default_color_primary_dark));
}
```

Tenemos 4 EditText en el aspecto, pues cojemos 4 editText y les asignamos el valor buscando con el identificador del EditText y llamamos al metodo cadaVez se realiza el foco al EditText del mismo, pasando la imagen del mismo.

```
public class MainActivity extends AppCompatActivity {

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.layout_add_new_contact);

        EditText editTextNombre = findViewById(R.id.editText);
        EditText editTextApellidos = findViewById(R.id.editText2);
        EditText editTextTelefono = findViewById(R.id.editText3);
        EditText editTextCorreo = findViewById(R.id.editText4);

        editTextNombre.setOnFocusChangeListener( (v, hasFocus) -> {
            onFocusChangeParaTodos(hasFocus, (ImageView) findViewById(R.id.imageView7));
        });
        editTextApellidos.setOnFocusChangeListener( (v, hasFocus) -> {
            onFocusChangeParaTodos(hasFocus, (ImageView) findViewById(R.id.imageView11));
        });
        editTextTelefono.setOnFocusChangeListener( (v, hasFocus) -> {
            onFocusChangeParaTodos(hasFocus, (ImageView) findViewById(R.id.imageView12));
        });
        editTextCorreo.setOnFocusChangeListener( (v, hasFocus) -> {
            onFocusChangeParaTodos(hasFocus, (ImageView) findViewById(R.id.imageView9));
        });
    }
}
```